HelenOS: State of the Union

Martin Děcký

DEPARTMENT OF DISTRIBUTED AND DEPENDABLE SYSTEMS
http://d3s.mff.cuni.cz/

CHARLES UNIVERSITY IN PRAGUE
FACULTY OF MATHEMATICS AND PHYSICS
What's New?

• **Since March 16\textsuperscript{th} 2010**
  
  • 572 days (1.56 years) ago
  
  • **Latest release 0.4.3**
    
    - March 26\textsuperscript{th} 2011
    
    - Last immediate release before 0.5
  
  • **On track to 0.5**
    
    - Next major release (the biggest ever)
    
    - Several major core improvements
      
      • Limitations in kernel memory management
    
    - Expected Q1 or Q2 2012
A Few Figures

- Physical lines of code
  - 144,625 → 288,432 +143,807 251 lines/day

- Development effort
  - Basic COCOMO model
    - 37.09 py → 76.57 py +39.48 py 25 persons
    - $5,010,686 → $10,343,970
    - 315 commits → 1269 commits
      - Not counting commits in merged branches
        +954 com. 1.66 com./day
### A Few Figures (2)

- **Language split**
  - Almost the same

<table>
<thead>
<tr>
<th>Language</th>
<th>Previously</th>
<th>Now</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>90.88 %</td>
<td>92.94 %</td>
</tr>
<tr>
<td>Assembler</td>
<td>5.67 %</td>
<td>2.98 %</td>
</tr>
<tr>
<td>Python</td>
<td>2.02 %</td>
<td>1.49 %</td>
</tr>
<tr>
<td>Shell</td>
<td>0.32 %</td>
<td>1.27 %</td>
</tr>
</tbody>
</table>
Achievements

• **GSoC 2011**
  - Our 3rd attempt
    - Learned from our own possible mistakes
    - Learned from others
    - A foot in the door
  - 20+ proposals
    - People all over the world
    - Spreading the word

http://helenos.org/go-gsoc
http://helenos.org/gsoc2011
http://google-opensource.blogspot.com/2011/10/google-summer-of-code-students-shine.html
• Accepted & passed projects

• Jiří Zarevúcký (mentored by Martin Děcký)
  - Bachelor student @ Faculty of Informatics, Masaryk University in Brno
  - Port of PCC, libposix

• Petr Koupý (mentored by Jiří Svoboda)
  - Masters student @ Faculty of Mathematics and Physics, Charles University in Prague
  - Port of GNU binutils, libposix

• Oleg Romanenko (mentored by Jakub Jermář)
  - Ph.D. Student @ Volgograd State Technical University (system analysis)
  - FAT12/16/32 with LFN, exFAT
• **Overall impressions**

  - Very good programmers, sticking to the schedule
    - Major goals accomplished half-way through
    - Easy to integrate into mainline thanks to frequent merging, non-intrusive way of programming
  
  - Right on track to full self-hosting
    - Simple binaries (e.g. tetris) can already be built within HelenOS
    - A few limitations of PCC (inline assembly)
    - Future port of GCC quite viable
Demo
• GSoC Mentor Summit 2011
  • Unconference
  • October 22\textsuperscript{nd} – 23\textsuperscript{rd} 2011
  • Googleplex @ Mountain View, California
  • Attended by Jiří Svoboda and Martin Děcký
    - Committed to a talk about the synergies of open source and academia
Achievements (2)

- **Team software projects**
  - **USB support**
    - October 2010 – June 2011
    - Matúš Dekánek, Vojtěch Horký, Matěj Klonfar, Šuboš Slovák, Ján Veselý
    - UHCI, OHCI (USB 1.1), EHCI stub, virtual USB
    - Initially for HID devices (keyboards, mice)
    - Integrated
    - Early UMS support
    - Ongoing development
Achievements (3)

- **Team software projects**
  - **Networking improvements**
    - January 2011 – September 2011
    - Zdeněk Bouška, Jiří Michalec, Radim Vansa, Jan Záloha
    - Generic device driver interface for network cards
      - Auto-negotiation, filtering, off-loading, etc.
      - General improvements in the networking stack
    - Novell NE2000, Realtek RTL 8319, Intel E1000
    - Partially integrated
  - Networking stack
    - TCP still not production quality
Achievements (4)

• Theses
  
  • **Martin Sucha:** Support for ext2 (read-only)
    - Bachelor thesis @ Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava
    - Supervised by RNDr. Jaroslav Janáček, Ph.D.
    - Integrated, ongoing development
  
  • Štěpán Henek: Security containers and access rights
    - RBAC security model and implementation
    - Not integrated yet
Achievements (5)

• Theses
  
  • Jiří Tlach: MMU-less operation
    
    – First open source implementation of XFI method
      
      • Originally by Microsoft Research
        Erlingsson et al.: XFI: Software Guards for System Address Spaces, OSDI, 2006
      
    – STATIF (STATic binary Instrumentation Framework)
      
      • Powerful ELF instrumentation tool using binary rewriting
      
      • Based on previous master thesis by Alexandr Kára
      
    – Not integrated yet
    
    – Publication potential
Achievements (6)

- **Theses**
  - **Vojtěch Horký**: NUMA support
    - Not integrated yet
  - **Stanislav Kozina**: Monitoring
    - System statistics (integrated)
    - Statistical profiler (not integrated yet)
  - **Tomáš Brambora**: Task snapshotting
    - Not integrated yet
  - **Lenka Trochtová**: Device driver framework
    - Fully integrated and extended
Achievements (7)

- **Theses currently in progress**
  - Muong Nam Han: Support for NTFS
  - František Princ: Support for ext4
  - Tomáš Benhák: Port to Xen hypervisor
  - Jan Dolejš: HelenOS as Xen hypervisor
  - Zdeněk Bouška: VFS-FUSE connector

- **Contemplated theses**
  - Object-oriented IPC interface (Objective-C)
  - Continuous integration tool
  - GUI stack
  - Sound stack
Achievements (8)

- Other improvements in the last 572 days
  - ABI separation, boot loader unification (deflate)
  - IA-64 revival (Itanium, Itanium II)
  - PowerPC revival
  - Neo FreeRunner support
  - i486 support
  - Memory reservations
    - Next-fit allocator, discontinuous areas
  - Support for Clang compiler
    - Autotooling
Achievements (9)

- **Other improvements in the last 572 days**
  - ATA: Support for 48-bit LBA
  - IPC: Safe client soft-state passing
  - Location service
  - Filesystem support
    - MINIX FS v3 (Maurizio Lombardi)
    - ISO 9660 (Jiří Kavalík, Martin Děcký)
    - libblock, libscsi
  - Support for GCC 4.5+ link-time optimization
  - Support for core dumps
Achievements (10)

• Other improvements in the last 572 days
  
  • Sysel (by Jiří Svoboda)
    - Multi-paradigm, object-oriented programming language
    - Bootstrap interpreter functional
    - Compiler work-in-progress
  
  • Improved async framework
    
    ```c
    async_sess_t *sess = async_connect_me_to(EXCHANGE.Serialize, ...);
    async_exch_t *exch = async_exchange_begin(sess);

    aid_t req = async_send_0(exch, PUT_DATA, NULL);
    sysarg_t rc = async_data_write_start(exch, (void *) data, size);

    async_exchange_end(exch);

    async_wait_for_(req, &rc);
    ```
Research prospects

- **Keywords**
  - **Correctness**
    - With respect to the composition of components
      - Correct sequencing of method calls, deadlock freedom
      - Substituability
    - With respect to the compliance between external behavior specification and implementation of components
      - Stubs, skeletons, connectors generation from specification
      - Run-time interface compatibility checking
    - With respect to internal behavior of components
      - Preconditions, postconditions, invariants
      - Safety properties, extra-functional properties
Keywords

- Manageability
  - Granularity of components
    - Isolation vs. performance
    - Separation of concerns at run-time
  - Change management
    - Separation of concerns at design-time
    - Encapsulation of internal behavior
  - Debugging
    - Non-intrusive monitoring and instrumentation
    - Mapping of run-time concepts to design concepts
Correctness

Semantic information in source code
Architecture and behavior specification
Architecture models
Extra-functional properties
Correctness

- Semantic information in source code
- Compiler checks
  - Static analyzers
  - Abstract interpretation
- Architecture and behavior specification
- Compatibility checks
  - Compliance checks
  - Code generation
- Architecture models
- Extra-functional properties
- Model checking
- Use case analysis
- Performance modeling
- Schedulability analysis
Correctness (2)

• **Means**

  • **Semantic information in source code**
    - Static analyzers
      • Stanse, Clang Static Analyzer, Coverity
      • Non-exhaustive, limited capabilities, but helpful
    - Built-in compiler checks
      • Uninitialized variables, unreachable code, etc.
      • Extended source code annotations
        `__attribute__((noreturn))`
    - Abstract interpretation tools
      • Frama-C (logical preconditions, postconditions, invariants)
      • VCC (race conditions, object ownership)
Correctness (3)

• Means

• Architecture and behavior description
  - Architecture description language
    • Hierarchical description of components interfaces and nesting
    • Stub and skeleton code generation
    • Connectors not needed at this point (single communication method)
  - Behavior protocols with custom extensions
    • Basic behavior protocols with tentative and alternative keywords
    • Macro preprocessing
    • Single-threaded semantics
    • Limited by the usefulness of behavior protocol checkers
Correctness (4)

- **Means**
  - **Architecture modeling**
    - Models of key synchronization algorithms
    - Checked by model checkers
  - **Extra-functional properties**
    - Not tackled yet
    - Future work
      - In-house performance models
      - Timed automata, DT & CT Markov chains, etc.
Current work

- **Fine semantic verification in compile-time**
  - **Motivation:** If code is incorrect, it should look suspicious or be syntactically wrong
    - Async framework
      - Sessions and exchanges as separate types
    - Kernel synchronization
      - Different spinlocks for interrupts disabled and enabled
    - Scalar types
      - Type system extending the run-time C types
      - Clang extension for scalar types
typedef unsigned int __attribute__((dimension("char"))) size_t;
typedef unsigned int __attribute__((dimension("1"))) count_t;
typedef unsigned int __attribute__((dimension("wchar"))) length_t;

extern length_t str_len(char *);
extern size_t str_size(char *);

char string[250];
string[str_size(string)] = 0; /* OK */
string[str_len(string)] = 0; /* Error (char array cannot be indexed by scalar wchar) */

#define PAGE_WIDTH 10
#define PAGE_SIZE (1 << PAGE_WIDTH)

typedef unsigned int __attribute__((dimension("uint8_t"))) uintptr_t;
typedef unsigned int __attribute__((dimension("uint8_t / PAGE_SIZE"))) vpn_t;

uintptr_t addr;
vpn_t page;

page = addr; /* Error (scalar types with different scale) */
page = addr >> PAGE_WIDTH; /* OK */
Conclusion

- Development efforts on multiple fronts
  - Plentitude of contributors and contributions
    - Some managing overhead, but lucky with several active maintainers
    - A semi-obligatory roadmap with features ahead

- Research efforts on multiple fronts
  - Mostly a one-man show with little outside input
    - Many things started, little finished (or even close to finish)
    - No detailed roadmap
1. Go to http://www.helenos.org/download
2. Fetch the ISO image for *amd64* or *ia32*
3. Optionally: Burn the image on a CD
4. **Boot the image or CD**
   - Any decent PC or virtual machine should do
5. **Play!**
6. Go to http://trac.helenos.org/
   - Get the sources from bzr://bzr.helenos.org/mainline
Ὤ ξεῖν', ἀγγέλλειν Λακεδαιμονίοις ὅτι τὴδε κείμεθα, τοῖς κείνων ρήμασι πειθόμενοι.

Go tell the Spartans, thou who passest by, that here, obedient to their laws, we lie.